UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

| APPLICATION NO. | FILING DATE | FIRST NAMED INVENTOR | ATTORNEY DOCKET NO. | CONFIRMATION NO. |
|---|---|---|---|---|
| 09/191,281 | 11/12/1998 | NEELAKANTAN SUNDARESAN | AM9-98-157 | 8452 |

22462 7590 01/10/2005

GATES & COOPER LLP
HOWARD HUGHES CENTER
6701 CENTER DRIVE WEST, SUITE 1050
LOS ANGELES, CA 90045

| EXAMINER | |
|---|---|
| SINGH, RACHNA | |
| **ART UNIT** | **PAPER NUMBER** |
| 2176 | |

DATE MAILED: 01/10/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

**BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES**

Application Number: 09/191,281
Filing Date: November 12, 1998
Appellant(s): SUNDARESAN, NEELAKANTAN

George H. Gates
For Appellant

**EXAMINER'S ANSWER**

This is in response to the appeal brief filed 6/28/04.

***(1) Real Party in Interest***

A statement identifying the real party in interest is contained in the brief.

***(2) Related Appeals and Interferences***

The brief contains a statement stating that there are no related appeals and interferences which will directly affect or be directly affected by or have a bearing on the decision in the pending appeal is contained in the brief.

***(3) Status of Claims***

The statement of the status of the claims contained in the brief is correct.

***(4) Status of Amendments After Final***

The appellant's statement of the status of amendments after final rejection contained in the brief is correct.

***(5) Summary of Invention***

The summary of invention contained in the brief is correct.

***(6) Issues***

The appellant's statement of the issues in the brief is correct.

***(7) Grouping of Claims***

Appellant's brief includes a statement that claims do not stand or fall together.

***(8) Claims Appealed***

The copy of the appealed claims contained in the Appendix to the brief is correct.

***(9) Prior Art of Record***

| | | |
|---|---|---|
| 6,272,673 | Dale | 8-2001 |
| 5,920,879 | Kyojima et al | 7-1999 |

Bray, "Extensible Markup Language (XML): Part 1. Syntax," http://www.w3.org/TR/WD-xml-lang-970331, March 31, 1997.
Kirsanov, "XML DTDs and Valid XML Documents," http://www.webreference.com/dlab/books/html/38-3.html , Jun. 16, 1997

Bray, "Document Content Description for XML," http://www.w3.org/TR/NOTE-dcd, July 31, 1998.

Softquad HotMetalPro 3.0 User's Manual, 1996, pages 77-83.

W3C Extensible Markup Language (XML) 1.0, 2/1998, available: http://www.w3.org/TR/1998/REC-xml-19980210.

***(10) Grounds of Rejection***

The following ground(s) of rejection are applicable to the appealed claims:

***Claim Rejections - 35 USC § 103***

1. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

2. **Claims 1, 4-7, 24, 27-30, 47, and 50-53 are rejected under 35 U.S.C. 103(a) as being unpatentable over Dale, U.S. Patent No. 6,272,673 B1, Aug. 7, 2001, filed Nov. 25, 1997 in view of Kyojima et al (5,920,879).**

Regarding independent claims 1, 24, and 47, Dale discloses "generating class specifications in the computer system," or **create one director component**, as in column 10, lines 50-59, as the components in questions are Java components embedded in the document. Dale teaches displaying a graphical user interface on a display device and using an editor to integrate components into a hypertext pages to

create an application. Dale's editor allows text to be combined with various components. See abstract and columns 19-20. Furthermore, Dale discloses "instantiating objects in the computer system from the class specifications," as in column 10, lines 60-63, as Dale describes **instantiating components**, particularly Java components as explained in column 5, lines 7-9.

It is noted that Dale does not explicitly say the **components** are "class specifications". However, refer to Kyojima's abstract and columns 1-4, in which he discloses the generation of "class specifications" from "schemas". It would have been obvious to one of ordinary skill in the art at the time of the invention to combine the inventions of Dale and Kyojima in arriving at the instant invention because such combination would facilitate adaptation of one type of component model to another according to structural constraints (see Kyojima's column 3, bottom). This would further aid Dale's invention in moving application components from tier to tier (see Dale's column 2).

Regarding claim 4, 27, and 50, Dale discloses that the class specifications are "Java class specifications," as in column 13, lines 1-3, where it is explained that the components and interface are actually a **specification of the methods, or subroutines, which a component supports** within the Java class.

Regarding claims 5, 28, and 51, Dale discloses the "generating step of converting an entity in the schema into the class specification," as in column 3, lines 64-67, where Dale describes the **CPU executing instructions which convert the received instructions**, already known to be Java components, **to instructions which**

**can be directly executed by the CPU**, which is known to be the instantiated objects from the class specification.

Regarding claims 6, 29, and 52, Dale discloses in column 20, 5-15 that the user can **customize and select which components are included in the application** which are "optional customization specifications."

Regarding claims 7, 30, and 53, Dale discloses custom specifications that define the class names to generate, as in column 6, lines 18-20, where the programs are like applets where the class names must be defined, as in the example in column 14, lines 57-60.

3. **Claims 2, 17-18, 25, 40-41, 48, and 63-64 are rejected under 35 U.S.C. 103(a) as being unpatentable over Dale in view of Kyojima, as applied to claims 1, 24, and 47 above, and further in view of Bray, "Extensible Markup Language (XML): Part I. Syntax," http://www.w3.org/TR/WD-xml-lang-970331, Mar. 31, 1997.**

Regarding claims 2, 25, and 48, a method for converting schemas to component models is disclosed above; however, Dale does not expressly disclose using XML documents and schemas. Bray discloses the use of XML syntax within documents and the schemas involved, as in page 7+.

At the time the invention was made, it would have been obvious to a person of ordinary skill in the art to combine Dale, Kyojima, and Bray to convert document schemas to component models where the documents are XML schemas. One of ordinary skill in the art would have been motivated to do this because it was known that XML was slowly becoming the primary standard for well-formed, definable, markup

languages, and also because of its ability to define the tags within the document it was very useful to specify objects and indicate specific components within the document.

Regarding claims 17-18, 40-41, and 63-64, Dale in view of Kyojima does not expressly disclose the use of regular expression languages. However, Bray discloses the use of a regular expression language using EBNF notation that comprises of the operators, and also optimizing the regular expression language, as on page 5. Furthermore, Bray discloses, under section D of page 38, that the non-deterministic models are **reduced** or "optimized."

At the time the invention was made, it would have been obvious to a person of ordinary skill in the art to generate class specifications from a regular expression language. One of ordinary skill in the art would have been motivated to do this because XML already clearly uses and interprets the EBNF standard. Also, it would be obvious to want to be able to instantiate multiple types of languages, and regular expression languages are a common type of language. Furthermore, at the time the invention was made, it would have been obvious to a person of ordinary skill in the art to optimize the regular expression language. One of ordinary skill in the art would have been motivated to do this because it would be more efficient to compile and run.

4. **Claims 3, 26, and 49 are rejected under 35 U.S.C. 103(a) as being unpatentable over Dale in view of Kyojima and Bray as applied to claims 2, 25, and 48 above, and further in view of Kirsanov, "XML DTDs and Valid XML Documents," http://www.webreference.com/dlab/books/html/38-3.html, Jun. 16,**

**1997 and Bray, "Document Content Description for XML," http://www.w3.org/TR/NOTE-dcd, July 31, 1998, now known as Bray 2.**

Regarding claims 3, 26, and 49, a method of converting XML document schemas to component models is disclosed above; however, neither Dale nor Bray expressly discloses the use of DTDs and DCDs. Kirsanov discloses using DTD's for XML documents, as in pages 1-3 and Bray 2 discloses using DCD's, as in page 2, "Introduction."

At the time the invention was made, it would have been obvious to a person of ordinary skill in the art to combine Dale, Kyojima, Bray, and Kirsanov to use XML DTDs and DCDs to select XML schemas. One of ordinary skill in the art would have been motivated to do this because it was known that XML DCDs and DTDs describe the format of the document and also were used by XML parsers to validate the document. Therefore, if one were to use XML schemas then one would need to pick the appropriate schemas and constraints defined by the DCDs and DTDs.

5. **Claims 8-10, 21, 31-33, 44-45, 54-56, and 67-68 rejected under 35 U.S.C. 103(a) as being unpatentable over Dale, U.S. Patent No. 6,272,673 B1, Aug. 7, 2001, filed Nov. 25, 1997 in view of Kyojima et al (5,920,879), as applied to claims 1, 24, and 47 above, and further in view of Softquad HotMetalPro 3.0 User's Manual, 1996, pages 77-83.**

In reference to claims 8, 31, and 54, Dale and Kyojima's system does not state a group comprising a visual editor class, a content implementation class, and handler class; however, HotMetalPro allows the user to use Java in the document editing

window and provide various applets for visual means as well as customizing elements. See pages 77-80. It would have been obvious to one of ordinary skill in the art at the time of the invention to provide any of the above mentioned groups to the editor of Dale and Kyojima since a class specification could relate to the visual editor, the components of XML schemas and initiators of the visual editor.

In reference to claims 9-10, 32-33, and 55-56, HotMetal Pro teaches defining the applet with attributes and parameters. The Parameters can indicate the value and name of the object. See pages 78-79. It would have been obvious to one of ordinary skill in the art at the time of the invention to map and select the elements and attribute of elements to the editor since the system of Dale and Kyojima and HotMetal Pro are concerned with providing class specifications to identify components of the editor.

In reference to claims 21, 44, and 67, HotMetal Pro teaches the use of Java for use in a document editing window. The user can choose a class file or drag and drop it into the document window. See pages 77-80. It would have been obvious to one of ordinary skill in the art at the time of the invention to utilize implementations for widgets in Dale and Kyojima's system since utilizing java class objects for special processing of the screen was well known at the time of the invention.

In reference to claims 22, 45, and 68, HotMetal Pro teaches defining the applet with attributes and parameters. The Parameters can indicate the value and name of the object. See pages78-79. It would have been obvious to one of ordinary skill in the art at the time of the invention to utilize different expression editors in the system of Dale and Kyojima since both are concerned with using class specifications and editors.

**6. Claims 11-15, 34-38, and 57-61 are rejected under 35 U.S.C. 103(a) as being unpatentable over Dale, U.S. Patent No. 6,272,673 B1, Aug. 7, 2001, filed Nov. 25, 1997, Kyojima et al (5,920,879), and Softquad HotMetalPro 3.0 User's Manual, 1996, pages 77-83, as applied to claims 10, 33, and 56 above, and further in view of W3C Extensible Markup Language (XML) 1.0, 2/1998, available: http://www.w3.org/TR/1998/REC-xml-19980210.**

In reference to claims 11, 34, and 57, it was well known at the time of the invention that declarations of attributes in a Document Type Definition of an XML document comprised mandatory, optional, and fixed values. See W3C Recommendation XML 1.0 pages, 18-21. It would have been well known in the art to combine the XML features with the system of Dale/Kyojima/Bray since all are concerned with a system dealing with class specifications.

In reference to claims 12-14, 35-37, and 58-60, while reading in the DTD of the XML document, the declarations of the attributes are carried through. Thus, when the user defines certain attributes, it takes the declarations into consideration. See W3C Recommendation XML 1.0, pages 18-21. It would have been well known in the art to combine the XML features with the system of Dale/Kyojima/Bray since all are concerned with a system dealing with class specifications.

In reference to claims 15, 38, and 61, W3C teaches that there are validating processors that read the DTD and parse entities referenced in the document. See page 31 of W3C Recommendation XML 1.0. It would have been well known in the art to

combine the XML features with the system of Dale/Kyojima/Bray since all are concerned with a system dealing with class specifications.

**7. Claims 16, 23, 34-37, 39, 46, 57-60, 62, and 69 are rejected under 35 U.S.C. 103(a) as being unpatentable over Dale, U.S. Patent No. 6,272,673 B1, Aug. 7, 2001, filed Nov. 25, 1997 in view of Kyojima et al (5,920,879), as applied to claims 1, 24, and 47 above, and further in view of W3C Extensible Markup Language (XML) 1.0, 2/1998, available: http://www.w3.org/TR/1998/REC-xml-19980210.**

In reference to claims 16, 39, and 62, W3C teaches that there are validating processors that read the DTD and parse entities referenced in the document. See page 31 of W3C Recommendation XML 1.0. It would have been well known in the art to combine the XML features with the system of Dale/Kyojima/Bray since all are concerned with a system dealing with class specifications.

In reference to claims 23, 46, and 69, W3C teaches validating processors that check the validity of the schema and parses entities referenced in documents. In providing validation for a schema, W3C is solving correctness, optimization, and aesthetic issues. Thus it would have been obvious to one of ordinary skill in the art at the time of the invention to utilize validation processors to optimize, correct, and solve aesthetic issues in the editor of Dale, Kyojima, and Bray since it was well known to validate schemas in order to utilize it for applications.

**8. Claims 19-20, 42-43, and 65-66 are rejected under 35 U.S.C. 103(a) as being unpatentable over Dal in vi w of Kyojima and Bray, "Ext nsible Markup**

**Language (XML): Part I. Syntax," http://www.w3.org/TR/WD-xml-lang-970331, Mar. 31, 1997, as applied to claim 18 above, and further in view of Softquad HotMetal Pro 3.0 User's Manual, 1996, pages 77-83.**

In reference to claims 19-20, 42-43, and 65-66 HotMetal Pro teaches defining the applet with attributes and parameters. The Parameters can indicate the value and name of the object. See pages78-79. It would have been obvious to one of ordinary skill in the art at the time of the invention to define widgets with class specifications that are associated with the operators and entities since it was well known to define that information for editors at the time of the invention as taught by HotMetal Pro.

***(11) Response to Argument***

In reference to claims 1, 24, and 47, Appellant argues that the complete limitation, "generating one or more class specifications in the computer from a schema in a document, wherein the class specifications identify user interface components of the editor corresponding to entities define in the schema" is not shown by Dale. Examiner disagrees. As stated above, Dale teaches creating one director component in which the components are Java components embedded in the document. See column 10, lines 50-59. Compare to "generating class specifications in the computer system". Dale also teaches displaying a graphical user interface on the display device and using an editor to integrate components into a hypertext pages to create an application. Dales' editor allows text to be combined with various components. See abstract and columns 19-20. Appellant argues that the editor in Dale merely constructs web-based applications, wherein the web pages include tags to applets that may be downloaded

when the pages are accessed, and the director component is merely an applet interconnects other components. Examiner disagrees. Dale explicitly teaches creating a director component and displaying a graphical user interface (GUI) on a display device an using an editor to integrate components into a hypertext page to create an application. See abstract. The "integration of components in an editor" teaches Appellant's claimed invention because it teaches that class specifications ((director components) identify user interface components in the editor. Examiner emphasizes that the limitation of "identifying user interface components of the editor" is taught by Dale in his disclosure of using an editor to integrate components (class specifications). See abstract and columns 19-20.

Appellant argues that Dale's editor is not generated from class specifications that are themselves generated from a schema for a document. Examiner states in rejections, "It is noted that Dale does not explicitly say the components are "class specifications". However, refer to Kyojima's abstract and columns 1-4, in which he discloses the generation of "class specifications" from "schemas". It would have been obvious to one of ordinary skill in the art at the time of the invention to combine the inventions of Dale and Kyojima in arriving at the instant invention because such combination would facilitate adaptation of one type of component model to another according to structural constraints. See Kyojima's column 3, bottom in which he recites, "There is provided a document structure composing apparatus for composing document structure which meets the structural constraint of a predetermined document class and characterized in that complementation specification storage means for storing the

element type of component which is required to complement a pre-complemented document structure composed according to a procedure, which does not meet the structure constraint of a first class, to adapt it to the first document class, and for storing a specification of complementation correlating to the element type and complementation means for analyzing said pre-complemented document structure based upon the structural constraint of the first document class. . ." Kyojima also discusses in the 'Background of the Invention' that one of the problems in the art at the time of the invention was that complementation in an area in document structure has an effect upon whether another area in document structure is suitable for the document class. The complementation to be performed in document structure not based upon only a specific local area but based upon the structure of the whole document. The second problem is that a user has no way to select one method of plural methods of complementing which are suitable for the document class. Thus Kyojima's invention is made to solve problems to provide a document structure composing apparatus to meet structural constraints of a predetermined document class and characterized in that complementation specification storage means for storing the element type of component which is required to complement a pre-complemented document structure composed according to a procedure, which does not meet the structure constraint of a first class, to adapt it to the first document class, and for storing a specification of complementation correlating to the element type and complementation means for analyzing said pre-complemented document structure based upon the structural constraint of the first document class."

Appellant argues that Kyojima does not teach or suggest class specifications generated from a schema. Examiner respectfully disagrees. As indicated in the rejections and comments above, Kyojima discloses the generation of "class specifications" from "schemas". See Kyojima's abstract and columns 1-4. It would have been obvious to one of ordinary skill in the art at the time of the invention to combine the inventions of Dale and Kyojima in arriving at the instant invention because such combination would facilitate adaptation of one type of component model to another according to structural constraints. See Kyojima's column 3, bottom in which he recites, "There is provided a document structure composing apparatus for composing document structure which meets the structural constraint of a predetermined document class and characterized in that complementation specification storage means for storing the element type of component which is required to complement a pre-complemented document structure composed according to a procedure, which does not meet the structure constraint of a first class, to adapt it to the first document class, and for storing a specification of complementation correlating to the element type and complementation means for analyzing said pre-complemented document structure based upon the structural constraint of the first document class. . ."

Appellant argues that Dale only describes components that are applets and does not teach that the user interface components of the editor are instantiated as objects to invoke the editor. Dale teaches that the editor can be used to integrate selected components into an extended hypertext page. In Dale's system a connection between

components of a set of hypertext pages is established and each of the components is referenced by at least one set of the pages.

Appellant generally argues with regards to the dependent claims that limitations are not taught in the same context as Appellant's claimed invention. Examiner disagrees with respect to the rejections provided above. All of the Applicant's limitations with regard to the dependent claims are taught by the references. See rejection.

Appellant argues with regards to claims 5, 28, and 51 that Dale does not teach converting an entity defined in the schema into a class specification. Dale discloses the CPU executing instructions which convert received instructions, already known to be Java components, to instructions which can be directly executed by the CPU, which is known to be the instantiated objects from the class specification. See column 3, lines 64-67.

Appellant argues with respect to claims 6, 29, and 52 that Dale does not teach generation class specifications from the schemas and one or more optional customization specifications. Examiner disagrees. Dale discloses in column 20, 5-15 that the user can **customize and select which components are included in the application** which are "optional customization specifications."

Appellant argues with respect to claims 7, 30, and 53 that Dale does not teach the optional customization specifications define what class names to generate for each entity. Examiner disagrees. Dale discloses custom specifications that define the class names to generate, as in column 6, lines 18-20, where the programs are like applets where the class names must be defined, as in the example in column 14, lines 57-60.

Appellant argues with respect to claims 8, 31, and 54 that HotMetalPro does not teach that the class specifications include one or more specifications from a group of a visual editor class, a content implementation class, and a handler class. While Dale and Kyojima's system does not state a group comprising a visual editor class, a content implementation class, and handler class, HotMetalPro allows the user to use Java in the document editing window and provide various applets for visual means as well as customizing elements. See pages 77-80. It would have been obvious to one of ordinary skill in the art at the time of the invention to provide any of the above mentioned groups to the editor of Dale and Kyojima since a class specification could relate to the visual editor, the components of XML schemas and initiators of the visual editor.

Appellant argues with respect to claims 9, 32, and 55 that HotMetalPro does not teach mapping entities defined in the schema to components in the editor. HotMetal Pro teaches defining the applet with attributes and parameters. The Parameters can indicate the value and name of the object. See pages 78-79. It would have been obvious to one of ordinary skill in the art at the time of the invention to map and select the elements and attribute of elements to the editor since the system of Dale and Kyojima and HotMetal Pro are concerned with providing class specifications to identify components of the editor.

Appellant argues with regards to claims 16, 39, and 62 that W3C does not teach that the class specifications include a function for validating at least one entity defined in the schema. Examiner disagrees. W3C teaches that there are validating processors

that read the DTD and parse entities referenced in the document. See page 31 of W3C Recommendation XML 1.0.

Appellant argues with regards to claims 17, 40, and 63 that Bray does not teach generating class specifications from a regular expression language. Examiner disagrees for reasons stated in the rejection above.

Appellant argues with regards to claims 18-19, 41-42, and 64-65 that regular expression language is not taught by Bray. Examiner disagrees for reasons stated in the rejection above.

Appellant's arguments with respect to claims 20-23, 43-46, and 66-69 have been addressed in the rejection above.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

JOSEPH FEILD
SUPERVISORY PATENT EXAMINER

RS
December 21, 2004

Conferees

Joseph Feild

Stephen Hong

GATES & COOPER LLP
HOWARD HUGHES CENTER
6701 CENTER DRIVE WEST, SUITE 1050
LOS ANGELES, CA 90045